CSLT TECHNICAL REPORT-20150020 [Friday 16th October, 2015]

Document Classification with Spherical Word Vectors

Yiqiao Pan[1*], Chao Xing[1] and Dong Wang[1]

[*]Correspondence: pyq@cslt.riit.tsinghua.edu.cn
[1]Center for Speech and Language Technology, Research Institute of Information Technology, Tsinghua University, ROOM 1-303, BLDG FIT, 100084 Beijing, China
Full list of author information is available at the end of the article

Abstract

Recent research shows that low-dimensional continuous representations of words (word vectors) can be successfully employed to classify documents, and document vectors derived from semantic clustering work better than those derived from simple average pooling. On the other hand, our recent study demonstrated that embedding words on a hypersphere offers better performance on tasks including semantic relatedness and bilingual translation when compared to the original approach that embeds words in an unconstrained plane space. In this paper, spherical word vectors are applied to the document classification task. The experiments show that spherical word vectors can deliver good performance when combined with semantic clustering based on vMF distributions.

Keywords: transfer learning; speech and language processing; deep learning

1 Introduction

Word embedding projects words into continuous low-dimensional representations, or word vectors. By this embedding, semantic or syntactic related words are located close to each other in the word vector space. The seminal research was conducted by Bengio and colleagues when studying neural language models [1], which has been followed by many researchers, e.g., [2][3][4][5].

Most of existing word embedding approaches embed words in *plane* space which means that there is no limit on the vector length. This means that the length if word vectors participates in representing word semantic and syntactic meanings. Ironically, when using word vectors to formulate semantic relations, the cosine distance was found to work better than the inner product. This means that the length information is essentially not useful, at least for semantic representation. Motivated by this observation, we proposed a spherical word embedding [6] which embeds words on a constrained hypersphere. With the spherical word vectors, the distance measure in the embedding phase (training) and the inference phase (semantic relation evaluation) becomes consistent. Our experiments have demonstrated that spherical word vectors outperform the original plane word vectors on a number of tasks including semantic relatedness prediction and bilingual translation [6].

In this paper, spherical word vectors are applied to the document classification task. In previous studies, we have shown that plane word vectors work well on this task, and outperform the poplar method based on latent Dirichlet allocation (LDA) [7]. In a subsequent study, we proposed a semantic clustering model, a distributional approach that derives document vectors from word vectors considering their distributions. This approach outperformed the conventional average pooling method that essentially ignores the distributional characteristic of word vectors. Our study in this paper shows that for spherical word vectors, the pooling approach simply fails, and with the distributional method, the performance can be significantly improved, even better than the semantic clustering method based on plain word vectors. Considering the property of spherical vectors, we propose a semantic clustering method based on the von-Mises-Fisher (vMF) mixture model and shows that a good performance can be achieved with the proposed method.

The rest of the paper is organized as follows: Section 2 introduces the related work; Section 3 describes the classical word embedding and semantic clustering; Section 4 presents spherical word vectors and the vMF-based semantic clustering. The experiments are presented in Section 5, and the paper is concluded by Section 6 .

2 Related work

Document classification has been a research focus for a long time. A typical classification system involves two components: document vector extraction and document classifier. The most popular document vector is based on word distributions, e.g., TF-IDF. This vector is large in dimensionality and does not consider semantic relations between words. Another type of document vector is based on various topic models, e.g., Latent Dirichlet Allocation (LDA) [8]. Our technique is inspired by the recent work on word embedding [5], which embeds words in a continuous low-dimensional space. In a previous study [7], we demonstrated that word vectors can be aggregated to derive document vectors by simple average pooling. In a subsequent research, the average pooling method was extended to a distributional method that leads to further performance improvement [9].

Another related work is the spherical word embedding method we proposed in [6]. Better performance was obtained with spherical word vectors on semantic relatedness and bilingual translation. This paper attempts to apply the new word representations to document classification. As we will see, simple average pooling works rather poor with spherical word vectors, and the distributional approach presented in [9] has to been employed.

Several studies have been conducted to model the distribution of spherical vectors, e.g., [10][11][12]. In this paper, the main mathematic tool is the von-Mises-Fisher (vMF) distribution [13], which is the simplest form for modelling spherical vectors, corresponding to the Gaussian distribution for plain vectors. Particularly, we are interested in the vMF mixture model which can describe complex spherical distributions, analog to the Gaussian mixture model (GMM) for plain vectors. This model has been studied by a multitude of researchers. For instance, Banerjee et al. [14] derived an Expectation Maximization (EM) procedure to estimate model parameters, and applied the model to text and genomic data clustering. In the limit, when the concentration parameter approaches infinity, the vMF mixture model reduces to the spherical k-means (SPKM) model [15]. Zhong [16] derived an online clustering method based on SPKM.

3 Word vectors and semantic clustering

3.1 Word embedding

Word representation is a fundamental problem in natural language processing. The conventional one-hot coding represents a word as a sparse vector of size $|V|$. This simple presentation is high-dimensional, discrete, and ignores semantic relations among words, which leads to much difficulty in model training and inference. An alternative approach is to represent words as low-dimensional continuous word vectors where relevant words are located to close to each other. The 'relevance' might be in the sense of semantic meanings, syntactic roles, sentimental polarities, or any others depending on the model objectives [5][14][17]. These continuous low-dimensional word representations are often called word vectors. Compared to the one-hot representations, word vectors posses significant advantage in semantic representation, model training and inference, as well as generalizability across domains and languages. Word vectors have attracted much attention and have attained remarkable success in a multitude of text processing tasks [18][19].

A popular word embedding approach is based on the skip-gram model proposed by Mikolov [20]. Basically, given a word and its left and context words in a particular sentence, the model tries update the word vectors so that the focused word is close to the context words in the embedding space, where the distance is measured by inner product. This model can be seen as a neural network where the input is the one-hot representation of the focused word w_i, denoted by e_{w_i}. This input one-hot code is projected to its word vector c_{w_i}, by looking up an embedding matrix U. This word vector, c_{w_i}, is then used to predict the word vectors of its left and right C neighbor words. Given a word sequence $w_1, w_2 ... w_N$, the training process maximizes the following objective function by optimizing the embedding matrix U which is composed of vectors of all the words in the vocabulary:

$$\mathcal{L}(U) = \frac{1}{N}\sum_{i=1}^{N}\sum_{-C \leq j \leq C, j \neq 0} logP(w_{i+j}|w_i)$$

where

$$P(w_{i+j}|w_i) = \frac{\exp(c_{w_{i+j}} c_{w_i})}{\sum_w \exp(c_w c_{w_i})}.$$

3.2 Semantic clustering

For document classification, it is essential to infer document vectors from the word vectors. A simple approach is to aggregate all the word vectors by average pooling. Surprisingly, this simple approach works very well in our experiments. A potential problem of this approach, however, is that the distributional information of words is totally ignored when deriving word vectors. A semantic clustering (SC) approach was proposed by the authors in [9]. In this approach, the word vectors of all the training documents are pooled to train a GMM, and then the vector of a document d is derived as the posterior probabilities that the document belongs to the Gaussian components of the GMM, formulated by:

$$v = [P(1|d), P(2|d), ..., P(M|d)]^T$$

where M is the number of Gaussian components of the GMM, and $P(k|d)$ is the posterior probability that d belongs to component k. Let j index all the words in the document, $P(k|d)$ is given by:

$$\begin{aligned}P(k|d) &= \frac{p(d|k)}{\sum_{r=1}^{M} p(d|r)} \\ &= \frac{\prod_{w_j \in d} p_k(c_{w_j})}{\sum_{r=1}^{M} \prod_{c_{w_j} \in d} p_r(c_{w_j})}\end{aligned} \tag{1}$$

and

$$p_k(c_{w_j}) = N(c_{w_j}; \mu_k, \Sigma_k)$$

where μ_k and Σ_k are the mean vector and covariance matrix of the k-th Gaussian component, respectively.

Compared to average pooling, the semantic clustering method infers semantic structures in the word vector space and then represents a document by these structures. Both the two steps rely on statistical learning, and thus leverages the distributional information of the words in the target document. Performance improvement was reported in [6]. This approach to document vector extraction is referred to as 'Gaussian SC' in this paper.

4 Document classification with spherical word vector

4.1 Spherical word vector

Spherical word vectors were proposed in [6]. The initial goal is to solve the inconsistency of the conventional word embedding methods in distance measure between the embedding phase and the inference phase. This new embedding approach is based on the conventional skip-gram model, but constrains the word vectors on the unit hypersphere. This can be achieved by solving a constrained optimization problem for example by Lagrange multipliers, though we adopted a simpler approach that normalizes the word vectors by the ℓ-2 norm whenever they are updated. Fig. 1 shows how conventional plane word vectors are regularized on the unit hypersphere. It has been shown that spherical word vectors can lead to better performance on tasks such as semantic relatedness and bilingual translation [6].

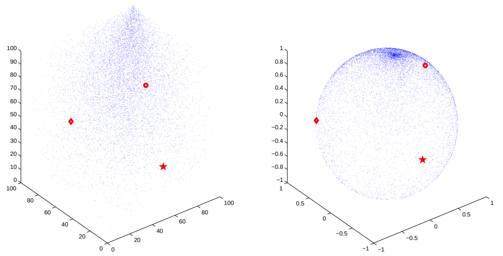

Figure 1: The distributions of plane and spherical word vectors. The red circles/stars/diamonds represent three words that are embedded in the two vector spaces respectively.

Applying spherical word vectors to document classification, however, is not as simple as with plane word vectors. Intuitively, simple average pooling seems not reasonable, because the mean vector is probably out of the unit hypersphere, which breaks the consistency between training and inference that that spherical word vectors were initially designed to achieve. We therefore resort to the distributional approach, i.e., semantic clustering, to derive document vectors.

4.2 vMF distribution for spherical vector

Semantic clustering for spherical vectors requires a suitable probabilistic model to represent the distributional characteristics of spherical data. The simplest form of distribution for spherical data is vMF, for which the probability density function is given by:

$$f_p(x;\mu,\kappa) = \frac{1}{Z_p(\kappa)} e^{\kappa \mu^T x} \tag{2}$$

where p is the dimensionality of data x, μ is the position parameter that satisfies $\|\mu\| = 1$, and $\kappa \geq 0$ is the concentration parameter. The partition function $Z_p(\kappa)$ is given by:

$$Z_p(\kappa) = \frac{(2\pi)^{p/2} I_{p/2-1}(\kappa)}{\kappa^{p/2-1}}$$

where I_v denotes the modified Bessel function of the first kind at order v. Note that the equations above apply for polar coordinates only.

4.3 Semantic clustering based on vMF mixture

Analog to Gaussian semantic clustering that generates document vectors from plane word vectors, vMF semantic clustering is derived to generate document vectors from spherical word vectors. The principle process is the same as the Gaussian SC presented in [9], except that the semantic clustering is based on the vMF mixture model instead of the Gaussian mixture model.

Specifically, suppose that the entire spherical space is represented by M mixtures of vMF distributions. Collect all the spherical word vectors of the training data (without considering boundary of documents), denoted by $\{c_i\}$. Train the vMF mixture model using these data, by optimizing the following objective function:

$$\mathcal{L}(\theta) = \prod_i \sum_{k=1}^{M} \pi_k f_p(c_i;\mu_k,\kappa_k)$$

where $\theta = \{\pi_k, \kappa_k, \mu_k : k = 1, 2, ..., M\}$ denotes all the parameters in the model. This optimization problem can be solved by the EM algorithm proposed by [14]. For simplicity, however, we choose the k-mean solution [15] that is much faster than the EM algorithm. Once the vMF mixture model has been trained, the vMF components provide a division of the entire spherical vector space, leading to a semantic clustering for spherical word vectors.

Similar to the Gaussian SC method for plane word vectors 3.2, a document d can be represented by the posterior probabilities that d belongs to the vMF components, i.e., $v = [P(1|d), P(2|d), ..., P(M|d)]$, and $P(k|d)$ is computed exactly the same as given by Eq.(1). The only difference is that the distribution is now a vMF instead of a Gaussian, given by:

$$p_k(c_{w_j}) = f_p(c_{w_j}; \mu_k, \kappa_k)$$

where $f_p(\cdot)$ is the probability density of the vMF distribution given by Eq.(2), and $\{\mu_k, \kappa_k : k = 1, 2, ..., M\}$ are the model parameters obtained with the k-mean training.

Once document vectors have been derived, a classifier can be used to perform document classification. This study chooses the support vector machine (SVM) with a linear kernel as the classifier, though any classifier is possible.

5 EXPRIMENTS

5.1 Databases

The experiments were conducted with two datasets: the Reuters dataset published by David D. Lewis[1] and the 20 Newsgroups dataset that was originally collected by Ken Lang[2].

The Reuters dataste is a collection of Reuters newswire in 1987. We use the LEWISSPLIT configuration, which uses 7337 documents for model training, and 3404 documents for test. There are 55 classes in total. For documents that are labelled by more than one topic, the first topic is chosen as the correct label.

The 20 Newsgroups dataset is a collection of approximately 20,000 newsgroup documents, evenly distributed across 20 newsgroups. We choose 80% of the documents for training and the rest for test.

5.2 Configurations

The word2vec tool provided by Google was used to produce conventional plane word vectors[3]. A simple modification mentioned in Section 4 was applied to the word2vec tool to produce spherical word vectors. The SVM model was built using the scikit-learn tool[4].

The two method for document vector derivation: average pooling and semantic clustering, are compared with each other. For plane word vectors, the sematic clustering is based on the GMM, and for spherical word vectors, it is based on the vMF mixture model. The GMM model was trained using the Weka toolkit[5], and the vMF mixture model was trained using an R implementation provided by Kurt Hornik[6].

5.3 Results

Table 1 presents the results on the Reuters dataset in terms of classification accuracy. For average pooling, the dimensionality of the document vectors is simply the same as that of the word vectors, which is fixed to 50 in our study. For semantic clustering, the dimensionality of document vectors corresponds to the number of Gaussian or vMF components, which varies from 50 to 200 in our experiment. The Gaussian semantic clustering (Gaussian SC) is applied to both plain and spherical word vectors, and vMF semantic clustering (vMF SC) is applied to spherical word vectors only.

[1] http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html

[2] http://kdd.ics.uci.edu/databases/20newsgroups/20newsgroups.html

[3] https://code.google.com/p/word2vec

[4] http://scikit-learn.org/dev/modules/svm.html#svm

[5] http://www.cs.waikato.ac.nz/ml/weka/

[6] http://cran.r-project.org/web/packages/movMF/index.html

The results shown in Table 1 demonstrate that for plane word vectors, average pooling works well and Gaussian SC offers marginal gains, when the number of Gaussian components is large enough. For spherical word vectors, average pooling works not as well, but semantic clustering delivers much more significant performance gains. Particularly, vMF SC outperforms Gaussian SC consistently. This is expected as vMF is more suitable to model spherical vectors. With vMF SC, spherical word vectors deliver better performance than plane word vectors.

We highlight that no matter how many components are used in semantic clustering, the basic information are all from the 50-dimensional word vectors. The additional gains obtained by semantic clustering with more Gaussian/vMF components are therefore totally attributed to the information conveyed by the distributional patterns of the words involved in a document. This again confirms the necessity of the distributional approach, e.g., the semantic clustering in our study.

Table 2 presents the results on the 20 Newsgroups dataset. Similar observations are obtained as in Table 1, except that semantic clustering here delivers much more significant performance gains, for both plane and spherical word vectors. Again, spherical word vectors with vMF SC offers the best performance.

Table 1: Classification accuracy on Reuters

		CA%	
	Dim	Plain WV	Spherical WV
Avg. Pooling	50	80.46	79.17
Gaussian SC	50	70.65	74.76
	100	77.47	79.96
	150	79.11	83.11
	200	81.14	82.17
vMF SC	50	-	76.26
	100	-	80.05
	150	-	81.52
	200	-	**83.84**

Table 2: Classification accuracy on 20 Newsgroups

		CA%	
	Dim	Plain WV	Spherical WV
Avg. Pooling	50	72.73	67.76
Gaussian SC	50	51.91	56.35
	100	74.11	74.32
	150	77.75	77.21
	200	80.53	80.99
vMF SC	50	-	56.81
	100	-	72.61
	150	-	78.24
	200	-	**81.73**

6 Conclusions

This paper applies spherical word vectors to the task of document classification. The experimental results confirmed that with vMF-based semantic clustering, spherical word vectors can be successfully applied to document classification, and achieve comparable or even better performance than conventional plane word vectors. In future work, we will investigate distributions more suitable for spherical word vectors.

Acknowledgement

This research was supported by the National Science Foundation of China (NSFC) under the project No. 61371136, and the MESTDC PhD Foundation Project No.20130002120011. It was also supported by Sinovoice and Huilan Ltd.

Author details

[1] Center for Speech and Language Technology, Research Institute of Information Technology, Tsinghua University, ROOM 1-303, BLDG FIT, 100084 Beijing, China. [2] Department of Computer Science and Technology, Tsinghua University, ROOM 1-303, BLDG FIT, 100084 Beijing, China.

References

1. Yoshua Bengio, Holger Schwenk, Jean-Sébastien Senécal, Fréderic Morin, and Jean-Luc Gauvain, “Neural probabilistic language models,” in *Innovations in Machine Learning*, pp. 137–186. Springer, 2006.
2. Andriy Mnih and Geoffrey E Hinton, “A scalable hierarchical distributed language model.,” in *NIPS*, 2008, pp. 1081–1088.
3. Joseph Turian, Lev Ratinov, and Yoshua Bengio, “Word representations: a simple and general method for semi-supervised learning,” in *Proceedings of the 48th annual meeting of the association for computational linguistics*. Association for Computational Linguistics, 2010, pp. 384–394.
4. Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa, “Natural language processing (almost) from scratch,” *The Journal of Machine Learning Research*, vol. 12, pp. 2493–2537, 2011.
5. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, “Efficient estimation of word representations in vector space,” *arXiv preprint arXiv:1301.3781*, 2013.
6. Chao Xing, Dong Wang, Chao Liu, and Yiye Lin, “Normalizedword embedding and orthogonal transform for bilingual word translation,” in *NAACL’15*, 2015.
7. Rong Liu, Dong Wang, and Chao Xing, “Document classification based on word vectors,” in *ISCSLP’14*, 2014.
8. David M Blei, Andrew Y Ng, and Michael I Jordan, “Latent dirichlet allocation,” *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.
9. Chao Xing, Dong Wang, Xuewei Zhang, and Chao Liu, “Document classification with distributions of word vectors,” in *Asia-Pacific Signal and Information Processing Association, 2014 Annual Summit and Conference (APSIPA)*. IEEE, 2014, pp. 1–5.
10. Christopher Bingham, “An antipodally symmetric distribution on the sphere,” *The Annals of Statistics*, pp. 1201–1225, 1974.
11. John T Kent, “The fisher-bingham distribution on the sphere,” *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 71–80, 1982.
12. Kanti V Mardia and Peter E Jupp, *Directional statistics*, vol. 494, John Wiley & Sons, 2009.
13. Nicholas I Fisher, *Statistical analysis of circular data*, Cambridge University Press, 1995.
14. Arindam Banerjee, Inderjit S Dhillon, Joydeep Ghosh, and Suvrit Sra, “Clustering on the unit hypersphere using von mises-fisher distributions,” in *Journal of Machine Learning Research*, 2005, pp. 1345–1382.
15. Inderjit S Dhillon and Dharmendra S Modha, “Concept decompositions for large sparse text data using clustering,” *Machine learning*, vol. 42, no. 1-2, pp. 143–175, 2001.
16. Shi Zhong, “Efficient online spherical k-means clustering,” in *Neural Networks, 2005. IJCNN’05. Proceedings. 2005 IEEE International Joint Conference on*. IEEE, 2005, vol. 5, pp. 3180–3185.
17. Andriy Mnih and Geoffrey Hinton, “Three new graphical models for statistical language modelling,” in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 641–648.
18. Tomas Mikolov, Quoc V Le, and Ilya Sutskever, “Exploiting similarities among languages for machine translation,” *arXiv preprint arXiv:1309.4168*, 2013.
19. Buzhou Tang, Hongxin Cao, Xiaolong Wang, Qingcai Chen, and Hua Xu, “Evaluating word representation features in biomedical named entity recognition tasks,” *BioMed research international*, vol. 2014, 2014.
20. Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean, “Distributed representations of words and phrases and their compositionality,” in *Advances in Neural Information Processing Systems*, 2013, pp. 3111–3119.